

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

Tinjauan pustaka ini yang akan menjadi landasan teori dalam penelitian yang dilakukan, merupakan pembahasan tentang penyelesaian masalah yang akan memberi jalan keluarnya, dalam hal ini akan dikemukakan teori-teori yang berkaitan dengan masalah yang diangkat.

2.2 Penelitian Terdahulu

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian penulis. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis:

Penelitian terdahulu yang pertama yang telah dilakukan oleh (Khotimah, Nafi'iyah, Masruroh, 2019) dengan judul “Klasifikasi Kematangan Buah Mangga Berdasarkan Citra HSV Dan KNN”. Berdasarkan uji coba yang telah dilakukan, dapat ditarik kesimpulan penerapan metode K-NN dapat diterapkan dengan baik dalam sistem klasifikasi kematangan mangga berdasarkan warna HSV. Akurasi yang didapatkan dari pengujian data testing memiliki rata-rata akurasi 55 % dengan jarak antara $k=1-10$. Menggunakan data training sebanyak 129 mangga dan data testing sebanyak 40 mangga dengan masing-masing kelas yakni kelas mentah, kelas cukup, kelas matang, dan kelas sangat matang. Sehingga Sistem klasifikasi kematangan buah mangga berdasarkan ekstraksi fitur warna HSV menggunakan metode *K-Nearest Neighbors* (KNN) tersebut layak untuk digunakan sebagaimana mestinya.

Penelitian yang kedua yang dilakukan oleh (Arif Patriot, Nafi'iyah, Qomariyah, 2019) dengan judul “ K-NN Klasifikasi Kematangan Buah Mangga Manalagi Menggunakan L*A*B dan Fitur Statistik”. Dalam penelitian ini membahas tentang hasil penelitian yang telah dilakukan dalam pengelompokan citra gambar buah mangga manalagi matang, setengah matang dan mentah dengan menggunakan K-Nearest Neighbor ini. Dari 130 data citra gambar yang terbagi menjadi 90 *data training* dan 40 *data testing*, kemudian 90 *data training* terdiri dari 40 citra gambar mentah, 10 citra gambar setengah matang dan 40 citra gambar matang sedangkan untuk 40 citra *data testing* terdiri dari 25 citra gambar buah mangga mentah, 5 citra gambar setengah matang dan 10 citra gambar matang.

Berdasarkan penelitian yang telah dilakukan oleh penulis, maka dapat diambil beberapa kesimpulan diantaranya adalah sebagai berikut:

1. Penerapan klasifikasi menggunakan L*A*B dan Fitur Statistik dapat diterapkan dengan baik dalam sistem klasifikasi kematangan buah mangga manalagi berdasarkan warna.
2. Ekstraksi ciri pada sistem klasifikasi kematangan buah mangga manalagi memperoleh nilai yang diambil yaitu nilai kontras, rata-rata, standar deviasi, korelasi, energi dan homogeniti.
3. Akurasi yang didapatkan dari pengujian data testing sebesar 62.5%.

Penelitian yang ketiga yang dilakukan oleh (Dendhi Restu Satri Utama, 2020) yang judul “Klasifikasi Kematangan Buah Mangga Menggunakan Metode SVM dengan Fitur HOG dan HVS” Peneliti menggunakan Data latih terdiri 16 citra untuk tiap variannya (total 80 citra untuk lima varian mangga). Data uji terdiri dari 4 citra untuk tiap variannya (total 20 citra untuk lima varian mangga). Data latih dan data uji didapatkan dengan cara difoto menggunakan kamera Smartphone 12mp jarak foto 20 cm, dan dengan latar putih. resize citra yang digunakan berukuran 240 x320 piksel pada bagian objek mangga saja. Research ini menggunakan fitur dari *Histogram of Oriented Gradients* (HOG) dan *Hue Saturation value* (HSV) sebagai input pelatihan dan pengujian metode

klasifikasinya dengan *Support Vector Machine* (SVM). Akurasi pengenalan kematangan buah mangga yang dapat dicapai dalam penelitian ini adalah sebesar 93%

Penelitian yang keempat yang dilakukan oleh (Kusuma, S., Pawening, R., & Dijaya, 2017). dengan judul. “ Otomatisasi Klasifikasi Kematangan Buah Mengkudu Berdasarkan Warna dan Tekstur”. Peneliti disini menggunakan algoritma *Support Vector Machine* (SVM) dan membandingkan dengan algoritma *K-Nearest Neighbors* (K-NN). Berdasarkan penelitian disimpulkan jika klasifikasi kualitas buah mengkudu dapat dilakukan menggunakan pengolahan citra digital. Klasifikasi paling baik diperoleh ketika menggunakan fitur warna yaitu 87.00% dan diklasifikasikan dengan algoritma SVM pada kernel linear. Uji coba pada kedua algoritma tersebut membuktikan algoritma SVM lebih unggul dibanding algoritma kNN. *Dataset* yang akan digunakan dibuat sendiri oleh peneliti. Pengambilan gambar dilakukan dengan bantuan *black box* yang telah dirancang khusus agar dapat mengurangi iluminasi cahaya yang ditimbulkan akibat *blitz* dari kamera yang diadopsi dari penelitian (Dijaya, Suciati, & Herumurti, 2016). *Dataset* yang digunakan berjumlah 60 data. Terdapat 20 data citra mengkudu berkualitas belum matang, 20 data citra mengkudu berkualitas matang, dan 20 data citra mengkudu berkualitas terlalu matang. Semua data akan dijadikan data uji menggunakan metode *10-cross validation*.

Penelitian yang kelima yang dilakukan oleh (Suastika Yulia Riska, Puji Subekti, 2016) dengan judul” Klasifikasi Level Kematangan Buah Tomat Berdasarkan Fitur Warna Menggunakan Multi-SVM “. Dengan Data yang digunakan pada penelitian ini sebanyak 108 citra tomat yang dibagi menjadi 66 data training dan 42 data testing. Citra tomat yang digunakan diambil dengan kamera smartphone dengan posisi tegak lurus terhadap objek.

Hal tersebut digunakan untuk menghindari adanya efek bayangan pada permukaan citra. Dataset yang digunakan pada penelitian ini merupakan citra buah tomat yang dapat dikelompokkan menjadi 6 level. Perbaikan citra pada tahap preprocessing dilakukan dengan adaptive histogram equalization. Pada feature extraction digunakan tiga fitur warna yang memiliki pengaruh terhadap nilai kematangan buah tomat, yaitu R, G, dan a*. Dari fitur warna yang digunakan, pada penelitian itu dilakukan pengujian untuk proses klasifikasi. Adapun proses klasifikasi level kematangan tomat dilakukan dengan membandingkan akurasi menggunakan metode Multi-SVM dan kNN. Pada tahap klasifikasi menggunakan Multi-SVM persentase yang diperoleh adalah 77,84% dengan fungsi kernel RGB. Sedangkan pada tahap klasifikasi kNN menggunakan k=3 dengan persentase 77,79%. Sehingga penggunaan metode multi-SVM dan KNN memiliki tingkat akurasi yang hampir sama, dengan nilai klasifikasi multi-SVM lebih tinggi dari pada kNN.

2.3 Contoh Perhitungan Penelitian Terdahulu

Pada penelitian ini proses ekstraksi fitur yang dipilih adalah ekstraksi fitur warna. Hal tersebut dikarenakan faktor yang mempengaruhi perbedaan level kematangan tomat adalah dari komponen warna yang terkandung di dalam citra tomat tersebut. Fitur warna yang digunakan pada penelitian ini meliputi komponen warna RGa*. Dalam perhitungan komponen warna citra tomat ditunjukkan pada Tabel 1.

Tabel. 1 Perhitungan komponen warna citra tomat

Citra Tomat	KOMPONEN WARNA		
	R	G	a*
1	70.55636	77.29844	119.2942
2	70.61289	77.05259	119.4905
3	70.96503	77.43419	119.3886
4	90.17973	82.6522	126.251
5	89.47845	81.70073	126.2519
6	90.38136	81.59172	126.7737
7	101.8091	77.92845	134.1216
8	104.0522	78.12156	135.042
9	104.1314	80.28676	134.0297

Tabel. 1 Lanjutan

Citra Tomat	KOMPONEN WARNA		
	R	G	a*
10	127.7628	51.49237	160.1941
11	128.3653	52.5377	159.9779
12	128.7216	52.32728	160.2202
13	115.2122	27.07452	165.3531
14	112.8548	29.4776	163.8833
15	115.0598	27.83456	165.178
16	102.9528	28.40958	161.0467
17	101.9537	30.79603	159.831
18	107.6799	33.22006	161.0903
19	82.90136	88.52844	131.4502
20	82.95789	88.28259	131.6465
21	83.31003	88.66419	131.5446
22	82.83752	88.86372	132.1324
23	83.15673	88.63826	131.8962
24	97.80973	90.77522	134.1452
19	82.90136	88.52844	131.4502
20	82.95789	88.28259	131.6465
21	83.31003	88.66419	131.5446
22	82.83752	88.86372	132.1324
23	83.15673	88.63826	131.8962
24	97.80973	90.77522	134.1452
19	82.90136	88.52844	131.4502
20	82.95789	88.28259	131.6465
21	83.31003	88.66419	131.5446
22	82.83752	88.86372	132.1324
23	83.15673	88.63826	131.8962
24	97.80973	90.77522	134.1452
26	98.01136	89.71472	134.6677
27	97.78453	89.73528	133.9863
28	97.64537	90.13748	134.2346
29	110.9326	87.05195	143.2451
30	113.1757	87.24506	144.1655
31	113.2549	89.41026	143.1532
32	112.3456	88.82528	143.6253
33	112.8634	89.12478	144.0614
34	135.4991	59.22865	167.9304

Tabel .1 Lanjutan

Citra Tomat	KOMPONEN WARNA		
	R	G	a*
35	136.1016	60.27398	167.7142
36	136.4579	60.06356	167.9565
37	135.8769	60.12738	167.8625
38	136.1117	60.12947	167.8354
39	122.5572	34.41952	172.6981
40	120.1998	36.82266	171.2283
41	122.4048	35.17956	172.5237
42	121.6884	34.96529	171.6884
43	121.9662	35.58263	172.4827
44	111.4849	36.94169	169.5788
48	114.6227	38.72527	169.2912
49	78.94099	82.93016	128.5443
50	78.99752	82.68431	128.7406
51	79.34966	83.06591	128.6387
52	98.56436	88.28392	135.5011
53	97.86308	87.33245	135.502
54	98.76599	87.22344	136.0238
55	110.1937	83.56017	143.3717
56	112.4368	83.75328	144.2921
57	112.516	85.91848	143.2798
58	136.1474	57.12409	169.4442
59	136.7499	58.16942	169.228
60	137.1062	57.95863	169.4703
61	123.5968	32.70624	174.6032
62	121.2394	35.10932	173.1334
63	123.4444	33.46628	174.4281
64	111.3374	34.04126	170.2968
65	110.3383	36.42775	169.0811
66	116.0645	38.85178	170.3404

Berdasarkan table 1, terdapat 66 data training dengan nilai fitur warna RGa*. Nilai fitur tersebut akan digunakan untuk proses klasifikasi dengan menggunakan Multi-SVM dan KNN.

Pada penelitian ini tahap klasifikasi level kematangan tomat menggunakan metode Multi-SVM dan KNN. . Penggunaan metode kNN

dalam pengujian level kematangan tomat menggunakan nilai $k=3$, $k=5$, $k=7$, $k=9$, dan $k=11$ yang ditunjukkan pada Tabel 2.

Tabel 2. Klasifikasi Level Kematangan Tomat

Level Tomat	k=3	k=5	k=7	k=9	k=11	Rata-
1	78.34	74.44	72.21	71.87	75.61	74.49
2	75.21	75.28	75.52	69.02	75.45	74.10
3	76.56	68.32	79.09	75.94	81.32	76.25
4	80.21	70.92	72.38	80.75	78.82	76.62
5	79.12	78.21	73.24	77.45	72.33	76.07
6	77.32	72.04	68.94	75.37	71.28	72.99
Rata-rata	77.79	73.20	73.56	75.07	75.80	75.09

Berdasarkan tabel 2 nilai rata-rata tertinggi adalah menggunakan $k=3$ yaitu sebesar 77.79%. Nilai akurasi tertinggi pada penggunaan $k=3$ adalah pada level 4, yaitu sebesar 80,21%.

Pada penelitian ini metode Multi-SVM terdiri dari 6 level kematangan tomat. Adapun fungsi kernel yang digunakan dalam penelitian ini antara lain, kernel linear, kernel radial basis gaussian (RBG), dan kernel polynominal. Dari ketiga kernel tersebut akan dibandingkan nilai akurasinya, sehingga diperoleh fungsi kernel terbaik untuk proses klasifikasi level kematangan tomat.

Berdasarkan Tabel 3 menunjukkan bahwa rata-rata tertinggi untuk klasifikasi level kematangan tomat adalah dengan menggunakan fungsi kernel radial basis Gaussian (RBG) sebesar 77.68%. Dari hasil klasifikasi level kematangan tomat tertinggi adalah pada level 6 menggunakan kernel RBG sebesar 81,34%. Bisa dilihat ditabel 3 berikut ini:

Tabel 3. Hasil Persentasi Klasifikasi Metode Multi-SVM

Level Tomat	Kernel (%)		
	Polynominal	RBG	Linear
1	80.32	79.56	68.31
2	75.54	78.33	65.29
3	73.23	75.95	60.33
4	74.43	77.68	61.26
5	75.04	74.18	58.45
6	73.28	81.34	59.22
Rata-rata	75.31	77.84	62.14

2.4 Klasifikasi

Klasifikasi merupakan sebuah metode yang digunakan untuk mengelompokkan data. Klasifikasi bisa juga diartikan sebagai pengelompokan data atau objek baru ke dalam suatu kelas atau kategori berdasarkan variabel-variabel tertentu. Klasifikasi memiliki teknik data mining yang melihat dari kelompok data yang sudah didefinisikan sebelumnya. Atribut ini digunakan sebagai variabel dalam menentukan kelas suatu objek yang baru. Klasifikasi ini memiliki tujuan yaitu menentukan kelas dari suatu objek yang kelasnya belum diketahui dengan akurat (nugraga, 2018).

Dalam proses klasifikasi terdiri dari dua fase, yaitu fase learning dan fase testing. Fase learning merupakan sebagian data yang kelas datanya telah diketahui sebelumnya dan dijadikan untuk model yang akan dibangun. Sedangkan fase testing memiliki arti fase model yang sudah terbentuk diuji dengan sebagian data lainnya untuk mengetahui akurasi dari model tersebut. Jika model akurasinya mencukupi, model ini bisa digunakan untuk memprediksi kelas data yang belum diketahui (Juhrah, 2015).

2.5 Citra

Menurut G. Sach dalam Soemirat dan Elviaro Ardianto (2007:171) citra bisa diartikan sebagai pengetahuan mengenai kita dan sikap-sikap terhadap kita yang mempunyai kelompok-kelompok yang berbeda. Penjelasan citra ini kemudian disitir oleh Effendi dalam Soemirat dan Elvinaro (2007:171) bahwa citra adalah dunia sekeliling kita yang memandang kita.

2.5.1 Jenis-jenis citra

Menurut seorang Frank Jefkins dalam Soemirat dan Elvinaro Ardianto (2007:117), memilah citra dalam beberapa jenis, antara lain:

- a. *The mirror image* (cerminan citra), yaitu bagaimana taksiran (citra) manajemen terhadap public eksternal dalam melihat perusahaannya.
- b. *The current image* (citra masih hangat), yaitu citra yang terdapat pada publik eksternal, yang berdasarkan pengalaman atau menyangkut

miskinnya informasi dan pemahaman publik. eksternal. Citra ini bisa saja berbenturan dengan mirror image.

- c. *The wish image* (citra yang diinginkan), yaitu manajemen mengharapkan perolehan prestasi tertentu. Citra ini diaplikasikan untuk sesuatu yang baru sebelum dipublikasikan secara eksternal untuk memperoleh informasi secara lengkap.
- d. *The multiple image* (citra yang berlapis), ialah sekelompok individu, kantor cabang atau perwakilan perusahaan lainnya dapat membentuk citra yang belum sesuai dengan kesamaan citra seluruh organisasi atau perusahaan.

2.5.2 Elemen-Elemen Citra Digital

Menurut seorang (Sutoyo, 2009:24) Citra Digital memiliki elemen-elemen sebagai berikut:

- a. *Kecerahan (Brightness)*
Mengambarkan kekuatan cahaya yang dikeluarkan melalui piksel dari citra sehingga dapat ditangkap oleh sistem penglihatan.
- b. *Kontras (Contrast)*
Merupakan komposisi terang dan gelap dalam sebuah citra, Citra yang baik memiliki komposisi gelap dan terang secara merata.
- c. *Kontur (Countour)*
Merupakan keadaan yang ditimbulkan oleh alterasi intensitas pada piksel-piksel yang berdekatan. Dengan adanya transisi intensitas inilah yang membuat mata mampu mendeteksi tepi-tepi objek didalam citra.
- d. *Warna (Colour)*
Warna sebagai persepsi yang bisa ditangkap sistem visual terhadap panjang gelombang cahaya yang dipantulkan melalui objek.
- e. *Bentuk (Shape)*
Merupakan properti intrinsik dari sebuah objek 3 dimensi, dengan pengetahuan bahwa bentuk merupakan properti intrinsik utama untuk sistem visual manusia.

f. *Tekstur (Texture)*

Identik sebagai penyaluran spesial dari derajat keabuan didalam sekumpulan piksel-piksel yang saling berdekatan.

2.5.3 Pengolahan Citra

Pengolahan citra merupakan suatu proses mengolah piksel-piksel didalam suatu citra digital yang bertujuan untuk mendapatkan hasil sesuai yang diinginkan. Membangun citra agar mudah ditafsirkan dan difahami oleh manusia. Citra dapat berupa gambar, video atau sesuatu yang sifatnya digital dan dapat dimasukan dalam memori penyimpanan. Setiap citra dapat diperoleh dengan cara akusisi citra, yaitu dimana proses yang dilakukan untuk mendapatkan suatu citra.

2.5.4 Teknik Pengolahan Citra

Menurut (Basuki, 2005:11) ada beberapa teknik dalam pengolahan citra yaitu sebagai berikut:

a. *Image Enhancement*

Suatu proses perbaikan citra dengan cara meningkatkan kualitas citra, baik kontras maupun kecerahan.

b. *Image restoration*

Suatu proses memperbaiki model citra, sehingga menjadi bentuk citra yang sesuai.

c. *Color Image Processing*

Suatu proses yang dikaitkan dengan citra warna, baik itu berupa image enhancement, image restoration, ataupun yang lainnya.

d. *Wavelet dan multiresolution processing*

Suatu proses citra yang menyatakan dalam berapa resolusi.

e. *Image Compression*

Suatu proses yang digunakan untuk mengubah ukuran data dalam suatu citra.

f. *Morphological Processing*

Proses untuk mendapatkan sebuah informasi yang menyatakan deskripsi suatu bentuk dari sebuah citra.

g. *Segmentation*

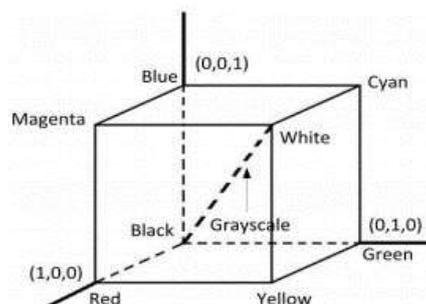
Proses untuk membedakan atau memisahkan objek-objek di dalam suatu citra, seperti memisahkan objek dengan background.

h. *Object Recognition*

Proses yang dilakukan untuk mengenali suatu objek apa saja yang ada didalam suatu citra.

2.6 Citra Warna RGB

Citra warna RGB merupakan suatu program yang digunakan untuk menampilkan warna di layar. RGB termasuk salah satu dari model warna. Adapun model gambar dengan model warna RGB memiliki tiga layer warna dasar pembentukan warna, yaitu red, green, blue. Secara matematis, warna RGB dimodelkan seperti sistem koordinat kartesius, dimana ketiga sumbunya bernama sumbu R, G dan B. Adapun pembagian ruang warna pada RGB divisualisasikan sebagai kubus seperti gambar dibawah ini. Pada gambar tersebut dapat diasumsikan semua nilai warna telah dinormalisasi oleh sistem sehingga rentang nilai yang dihasilkan adalah $[0,1]$. Nilai warna RGB yang utama terdapat pada sudut utama sumbu koordinat. Sedangkan sudut-sudut kubus selain kubus utama, merepresentasikan bahwa nilai warna sekunder yang dihasilkan adalah perpaduan warna utama. Warna sekunder tersebut, yaitu magenta, cyan, dan yellow. Di titik pusat koordinat $(0,0,0)$, merepresentasikan warna hitam, sedangkan warna putih terletak di sudut kubus yang letaknya paling jauh dari titik nol atau di koordinat $(1,1,1)$. Diagonal ruang yang ditarik dari $(0,0,0)$ sampai $(1,1,1)$, merupakan rentang nilai yang akan menghasilkan warna gradasi keabuan (Pamungkas, 2015). Berikut merupakan gambar citra warna RGB.



Gambar 2.1 Gambar Citra Warna RGB

2.7 Fitur *Grayscale*

Grayscale Merupakan citra yang hanya memiliki warna tingkat keabuan. Penggunaan fitur warna *grayscale* dikarenakan hanya membutuhkan sedikit informasi pada setiap piksel dibandingkan dengan citra berwarna. Warna abu-abu pada citra *grayscale* adalah warna R (*red*), G (*green*), dan B (*blue*) yang memiliki intensitas yang sama. Sehingga citra *grayscale* hanya membutuhkan nilai intensitas tunggal daripada citra berwarna yang membutuhkan tiga intensitas untuk setiap pikselnya. Intensitas dari citra *grayscale* akan disimpan dalam 8bit integer yang nantinya memberikan 256 kemungkinan yang mana dimulai dari level 0 sampai dengan 255 (0 untuk hitam dan 255 untuk putih dan nilai diantaranya adalah derajat keabuan). Tingkat keabuan atau *Grayscale level* (Anggraini, Hidayat, & Darana, 2017).

2.8 Segmentasi Citra

Segmentasi citra ini adalah salah satu bagian dari pengolahan citra dimana proses yang dilakukannya mengenalkan suatu objek ke sistem. Mereka juga mempunyai maksud untuk membelah citra jadi beberapa wilayah yang jenisnya dikelompokkan berdasarkan tingkatan keabuan dari *pixel* dengan tingkatan yang ada disebelahnya. Lalu dalam prosesi segmentasi suatu citra nantinya dilanjutkan ke proses pada tingkatan yang lebih tinggi terhadap suatu citra. Berikut ini merupakan proses segmentasi menurut Giannakopoulos (2008).

- a. Tentukan citra RGB yang menjadi objek deteksi, nilai warna *grayscale* yang menjadi acuan (hasil proses pelatihan data) dan nilai toleransi *grayscale* yang digunakan.
- b. *Transpose* citra RGB ke *grayscale*.
- c. Lakukan filter warna pada citra berdasarkan nilai acuan (T) dan nilai toleransi (tol). Dengan x sebagai warna *grayscale* pada piksel yang ada maka warna yang tidak termasuk dalam rentang $T - tol < x < T + tol$ diberi warna hitam.

- d. *Transpose* kembali citra ke RGB, tampilkan hasil filter (Nur Khasanah, Harjoko, & Candradewi, 2016).

2.9 *Algoritma Support Vector Machine (SVM)*

Algoritma SVM menyelesaikan masalah pengklasifikasian dengan cara mencoba untuk mencari pemisah *hyperplane* yang optimal antar kelas. *Hyperplane* bergantung dari kasus pelatihan yang meletakkannya pada tepi dari deskriptor kelas yang disebut *support vector* (Elhariri, et al., 2014). Prinsip dasar SVM adalah *linear classifier*, dan selanjutnya dikembangkan agar dapat bekerja pada masalah non-linear, dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi (Nursalim & Himawan, 2014). SVM juga dikenal sebagai teknik pembelajaran mesin (*machine learning*) paling mutakhir setelah pembelajaran mesin sebelumnya yang dikenal sebagai *Neural Network* (NN), baik SVM maupun NN tersebut telah berhasil digunakan dalam pengenalan pola (Nursalim & Himawan, 2014). Pembelajaran dilakukan dengan menggunakan pasangan data *input* dan data *output* berupa sasaran yang diinginkan atau biasa disebut pembelajaran terarah (Pawening, Dijaya, Brian, & Suciati, 2015). Pembelajaran terarah ini akan menghasilkan fungsi yang dapat menggambarkan bentuk ketergantungan *input* dan *output*-nya, harapannya fungsi tersebut akan mampu menggeneralisasi data *input* di luar data pembelajaran (Pawening, Dijaya, Brian, & Suciati, 2015). Secara matematika, konsep dasar SVM dijelaskan pada Persamaan 13,

$$\text{Min} \frac{1}{2} \|w\|^2 \text{ s.t. } y(xt \cdot w + b) - 1 \geq 0 \quad (13)$$

di mana $(xt \cdot w + b) \geq 1$ untuk kelas 1 dan $(xt \cdot w + b) \leq -1$ untuk kelas 2, xt adalah *dataset*, yt adalah *output* dari data $xt \cdot w$, b adalah parameter yang dicari.

2.10 *K-Nearest Neighbors (KNN)*

K-Nearest Neighbors adalah satu dari sekian banyaknya metode yang mempunyai fungsi mengerjakan dan menyelesaikan suatu pengelompokan terhadap suatu objek yang masukannya disesuaikan dari jarak yang mendekati

objek itu sendiri. Dari data yang dihasilkan nanti maka akan diproyeksikan kedalam ruang dimensi paling banyak dan dari situlah setiap dimensi mempunyai presentasi berbagai macam konfigurasi warna. Dalam ruang nanti akan diproses untuk dibagi lebih dari satu dengan pengelompokan dan pembenahan. KNN sendiri merupakan metode *supervised*, dimana hasil data dari query instance nanti akan dikelompokan berdasarkan dari banyaknya kategori metode tersebut.

K-Nearest Neighbors sendiri mempunyai tujuan yakni mengklasifikasikan obyek baru berdasarkan dari jarak suatu obyek yang akan diklasifikasikan dengan contoh data. Classifier ini menggunakan fungsi dari sebuah jarak data baru ke data training. *K-Nearest Neighbors* sendiri memiliki kepercayaan mencari sebuah jarak terdekat diantara data yang akan dievaluasi dengan data nilai K tetangga (*Neighbor*) terdekat dalam data pelatihan.

Menurut (Paramita, Cinantya, 2019) data pelatihan sendiri diproyeksikan keruang yang mempunyai dimensi banyak, dimana nanti masing-masing dimensi akan merepresentasikan fitur dari sebuah data. Ruang ini juga dibagi menjadi beberapa bagian berdasarkan klasifikasi dan pelatihan. Suatu titik yang ada diruang ini ditandai dengan kelas c, jika kelas c adalah klasifikasi yang paling banyak ditemui pada nilai k tetangga (*neighbor*) terdekat dari titik tersebut. Jauh atau dekatnya tetangga bisa dihitung berdasarkan dengan jarak euclidean dengan rumus 2.4 sebagai berikut:

$$d_i = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2} \dots\dots\dots 2.1$$

Keterangan:

x1 = Sampel Data

d= Jarak

x2 = Data Uji / Testing

p= Dimensi Data

i = Variabel Data

Pada saat edukasi, algoritma ini hanya mampu melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data edukasi. Pada saat klasifikasi ini, fitur-fitur yang sama dihitung untuk data test (yang klasifikasinya belum diketahui). Jarak dari vektor baru terhadap keseluruhan data vektor edukasi dihitung, dan

beberapa nilai k yang paling dekat diambil. Titik klasifikasi yang baru akan diprediksikan, termasuk pada klasifikasi terbanyak dari titik-titik tersebut.

Dalam algoritma ini nilai k tetangga (*Neighbor*) bergantung pada data, dimana nilai k yang tinggi akan mengurangi efek noise pada proses klasifikasi tetapi bisa membuat batasan antara setiap klasifikasi menjadi kabur. Pemilihan nilai k yang bagus dilakukan dengan cara optimasi parameter, seperti dengan menggunakan cross-validation. Kasus dimana klasifikasi diprediksi berdasarkan data pembelajaran yang paling dekat (dengan kata lain $k=1$) disebut algoritma nearest neighbor.

Menurut (Herdini:2019) ketepatan algoritma *K-Nearest Neighbor* ini sangat dipengaruhi dengan ada atau tidaknya fitur yang tidak relevan, atau jika bobot fitur tersebut tidak sama/setara dengan relevansinya terhadap klasifikasi. Ketika jumlah suatu data mendekati tak terhingga maka algoritma ini menjamin error rate yang tidak lebih dari dua kali bayes error rate.

2.11 *Microsoft Visio*

Microsoft Visio merupakan suatu program aplikasi komputer yang sering digunakan untuk membuat diagram atau perancangan, diagram alir (*flowchart*), *brainstorm*, dan skema jaringan yang diterbitkan oleh *Microsoft Corporation*. Dalam pembentukan suatu diagram aplikasi *Microsoft Visio* menggunakan grafik vektor untuk membentuk diagram-diagramnya. *Microsoft Visio* sebenarnya merupakan buatan *Visio Corporation* bukan buatan *Microsoft Corporation*, tapi pada tahun 2000 *Microsoft Visio* ini di akuisisi oleh *Microsoft*. Untuk versi pertama kali yang keluar menggunakan nama *Microsoft Visio* ialah *Visio 2002*, *Visio 2003* dan *Visio 2007*.

Dalam buku yang dituliskan oleh Helmers, (2013) yang berjudul *Microsoft Visio 2013 Step by Step*, Mengatakan bahwa: *Microsoft Visio* dapat diartikan sebagai aplikasi paling penting dalam pembuatan keseluruhan diagram bisnis, mulai dari *flowchart*, *network diagram*, *organization charts*, membuat denah dan *brainstroming diagram*.

2.12 *Flowchart*

Flowchart merupakan suatu bagan dengan simbol-simbol tertentu yang berfungsi menjelaskan dan menggambarkan suatu proses secara detail dan keterkaitan setiap prosesnya pada suatu program. Sedangkan menurut Indraji (2011) *flowchart* merupakan suatu gambaran grafik dari setiap proses dan juga urutan prosedur disetiap program. Adapun beberapa jenis *flowchart* yaitu sebagai berikut:

- ***Flowchart Sistem (SystemFlowchart)***

Flowchart Sistem ialah bagan yang menampilkan alur kerja suatu proses yang dilakukan di dalam metode secara keseluruhan dan menjelaskan urutan dari setiap proses yang ada didalam sebuah metode.

- ***Flowchart Dokumen (Document Flowchart)***

Flowchart Dokument ialah bagan alir yang memberi tahu alur dari laporan dan formulir beserta terusnya.

- ***Flowchart Skematik (Schematic Flowchart)***

Flowchart skematik merupakan bagan alir yang mirip dengan bagan alir sebuah sistem, yang berfungsi untuk menggambarkan setiap proses didalam sistem. bagan alir skematik selain menggunakan simbol yang sama dengan bagan alir sistem, juga menggunakan simbol-simbol, gambar-gambar komputer dan juga tools lain yang digunakan pada bagan alir sistem.

- ***Flowchart Program (Program Flowchart)***

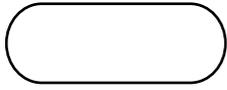
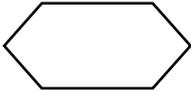
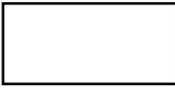
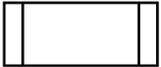
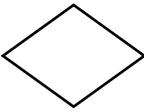
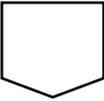
Flowchart program/bagan alir program merupakan suatu bagan yang menggambarkan secara detail prosedur dari setiap proses pada program. Bagan alir program dibuat dari hasil verifikasi pada bagan alir sistem.

- ***Flowchart Proses (Process Flowchart)***

Flowchart Proses merupakan teknik menggambar rekayasa industrial dengan cara memecah dan menganalisis proses selanjutnya dalam suatu prosedur atau sistem. *Flowchart* Proses atau Bagan alir proses menggunakan 5 simbol tersendiri.

- Simbol-Simbol dan Fungsi Flowchart

Tabel 2.1 Simbol dan Fungsi Flowchart

Simbol	Nama	Fungsi
	<i>Terminator</i>	Awalan / Akhir program
	Garis Alir (<i>Flow Line</i>)	Arah aliran arogram
	Preparation	Proses inisialisasi (pemberian harga awal)
	Proses	Proses Pengolahan Data
	<i>Input/Output</i> Data	Proses <i>Input/Output</i> Data
	<i>Predefined Proses</i> (<i>Sub Program</i>)	Proses awal menjalankan sub program
	<i>On Page</i> <i>Connector</i>	Penghubung bagian flowchart pada satu halaman
	<i>Decision</i>	Membandingkan data untuk memberikan pilihan lanjut langkah selanjutnya
	<i>Off Page</i> <i>Connector</i>	Menghubungkan bagian flowchart yang berada dihalaman berbeda

2.13 Matlab

Matlab merupakan singkatan dari *Matrices Laboratory*, Matlab sendiri dikembangkan oleh *MathWork* dan masuk dalam bahasa pemrograman tingkat tinggi. Matlab dikembangkan sebagai bahasa pemrograman dan sekaligus juga sebagai alat visualisasi yang memiliki kemampuan untuk menyelesaikan beberapa

kasus yang berhubungan langsung dengan keilmuan Matematika, seperti dibidang rekayasa teknik, fisika, statistika, komputasi dan juga modeling. Pengguna dapat memanfaatkan matlab untuk melakukan analisis data, mengembangkan algoritma dan juga membuat model maupun aplikasi. Dengan bahasa, tools, dan fungsi-fungsi built-in membuat pengguna lebih mudah dalam mengeksplorasi berbagai pendekatan sehingga memperoleh solusi lebih cepat dibandingkan dengan menggunakan spreadsheets atau bahasa pemrograman tradisional seperti C/C++ atau JavaTM.

Sebagai standar *variabel elemen*, *Matrices Laboratory* atau *Matlab* menggunakan konsep *Array/Matriks* tanpa memerlukan pendekatan *array* seperti bahasa pemrograman lainnya. Array sendiri merupakan sekumpulan data yang diatur dalam baris dan kolom yang diwakili oleh sebuah nama. Setiap nilai data dalam sebuah array dapat diakses dengan cara memasukan nama yang sesuai dengan *array* tersebut beserta posisi array dalam baris berapa dan kolom berapa. Dalam ruang lingkup pendidikan matlab menjadi alat pemrograman standart dalam bidang Matematika dan Rekayasa. Sedangkan dalam ruanglingkup industri matlab sendiri menjadi pilihan paling produktif yang digunakan untuk riset, komputasi dan analisa.

Matlab sendiri memiliki beberapa keunggulan dibandingkan dengan bahasa pemrograman lain yaitu sebagai berikut:

• **Keunggulan *Matlab* yaitu sebagai berikut:**

1. *Matlab* handal dalam komputasi yang terkait dengan array ataupun matriks. Keandalan dalam proses komputasi ini dapat dilihat dari tidak diperlukanya pendefinisian ukuran array/matriks dalam sebuah variabel. Ukuran atau dimensi dari variabel bertipe array secara otomatis akan menyesuaikan dengan array yang sudah ditetapkan dengan catatan ukuran variabel bertipe array ini sifatnya dinamis,
2. *Matlab* merupakan Platform yang berdiri sendiri.
3. Mempunyai fungsi-fungsi bawaan (*Predefined Function*).
4. Memiliki perangkat yang independen untuk menampilkan grafik maupun gambar.

5. Berbasis GUI (*Graphical User Interface*).

• **Matlab memiliki karakteristik yaitu sebagai berikut:**

1. Bahasa pemrograman yang berdasarkan pada matriks (Baris dan Kolom).
2. Menyediakan banyak toolbox yang digunakan dalam aplikasi-aplikasi khusus seperti: *Simulink*, *Neural Network*, *State Flow*, *Data Acquisition Toolbox*, *Image Acquisition Toolbox*, *Signal Processing Blockset*, *Communications Blockset*, *Fuzzy Logic* dan lain sebagainya.
3. Waktu pengembangan program matlab lebih cepat dibandingkan dengan pemrograman lain seperti Fortran dan C/C++.
4. Dalam penulisan source code program, tidak harus mendeklarasikan array terlebih dahulu.

2.14 Buah Sawo

Sawo atau tanaman buah dengan nama latin *Manikara zapota* merupakan tanaman tahunan yang bisa berbuah tanpa mengenal musim. Tanaman sawo ini berasal dari Amerika Tengah, tepatnya Meksiko dan India Barat. Namun saat ini, tanaman sawo sudah banyak tersebar luas di daerah tropik, termasuk Indonesia.

Sawo ialah salah satu tanaman yang cocok dibudidayakan di daerah tropis terutama di daerah dataran rendah. Maka tidak heran jika sawo banyak sekali dibudidayakan dan menjadi salah satu mata pencaharian oleh masyarakat Indonesia



Gambar 2.2 Gambar Buah Sawo

Berikut ini ialah klasifikasi dari tanaman sawo:

Tabel 2.2 Klasifikasi Buah Sawo

Kingdom	Plantae
Sub Kingdom	Viridiplantae
Infra Kingdom	Streptophyta
Super Devisi	Embryophyta
Devisi	Tracheophyta
Sub Devisi	Spermatophytina
Kelas	Magnoliopsida
Super Ordo	Asteranae
Ordo	Ericales
Famili	Sapotaceae
Genus	Manilkara Adans
Spesies	Manilkara zapota (L.) P. Royen

- **Buah Sawo Manila**

Sawo manila atau *Manilkara zapota* ialah pohon buah yang bertangkai pendek, bulat telur atau jorong, dengan ukuran 3-6 x 3-8 cm. Buahnya berwarna coklat merah sampai kekuningan, di bagian luar terdapat sisik-sisik kasar coklat yang mudah mengelupas. Buah sawo manila memiliki kulit tipis, dengan daging buah yang lembut dan terkadang memasir, memiliki rasa manis dan mengandung banyak sari buah. Ciri dari buah sawo manila lainnya ialah:



Gambar 2.1 Gambar Jenis sawo Manila

- Memiliki biji sampai 12 butir yang bentuknya lonjong pipih, hitam atau kecoklatan mengkilap.
 - Memiliki pohon dengan ukuran hingga 30 meter dengan batang berkulit kasar abu-abu kehitaman atau coklat tua.
 - Memiliki daun yang sedikit berbulu, berwarna hijau tua mengkilap, bentuk telur jorong sampai agak lanset, dengan ukuran 1,5-7 x 3,5-15 cm.
 - Memiliki bunga tunggal di ujung ranting, sisi luarnya memiliki bulu kecoklatan. Umumnya kelopak bunganya tersusun dari mahkota bentuk genta, berwarna putih
- **Klasifikasi Ilmiah Tanaman Sawo Manila**

Tabel 2.4 Klasifikasi Buah Sawo Manila

Kingdom	Plantae
Devisi	Magnoliophyta
Kelas	Magnoliopsida
Ordo	Ericales
Famili	Sapotaceae
Genus	Manilkara
Spesies	Manilkara zapota
Nama Binomial	Manilkara Zapota
Sinonim	Achras zapota L.